

Programowanie Zawansowane 2

Generated by Doxygen 1.15.0

1 Class Index	1
1.1 Class List	1
2 File Index	3
2.1 File List	3
3 Class Documentation	5
3.1 BSTree Class Reference	5
3.1.1 Detailed Description	6
3.1.2 Constructor & Destructor Documentation	6
3.1.2.1 BSTree()	6
3.1.2.2 ~BSTree()	6
3.1.3 Member Function Documentation	6
3.1.3.1 dodaj()	6
3.1.3.2 usun()	6
3.1.3.3 wyczysc()	7
3.1.3.4 wyswietl_inorder()	7
3.1.3.5 wyswietl_postorder()	7
3.1.3.6 wyswietl_preorder()	7
3.1.3.7 wyswietlGraficznie()	7
3.1.3.8 zapiszDoTekstowego()	7
3.1.3.9 znajdzSciezke()	8
3.2 FileManager Class Reference	8
3.2.1 Detailed Description	8
3.2.2 Member Function Documentation	9
3.2.2.1 wczytajBinarnie()	9
3.2.2.2 wczytajTekstowyDoDrzewa()	9
3.2.2.3 zapiszBinarnie()	10
4 File Documentation	11
4.1 BSTree.cpp File Reference	11
4.1.1 Detailed Description	11
4.2 BSTree.h File Reference	11
4.2.1 Detailed Description	11
4.3 BSTree.h	12
4.4 FileManager.cpp File Reference	13
4.4.1 Detailed Description	13
4.5 FileManager.h File Reference	13
4.5.1 Detailed Description	13
4.6 FileManager.h	13
4.7 main.cpp File Reference	14
4.7.1 Detailed Description	14
4.7.2 Function Documentation	15

4.7.2.1 main()	15
4.7.2.2 op_dodaj()	15
4.7.2.3 op_usun()	15
4.7.2.4 op_wczytajBin()	15
4.7.2.5 op_wczytajTekst()	16
4.7.2.6 op_wyczysc()	16
4.7.2.7 op_wyswietl()	16
4.7.2.8 op_zapiszBin()	17
4.7.2.9 op_zapiszTekst()	17
4.7.2.10 op_znajdzSciezke()	17
4.7.2.11 pobierzLiczbe()	18
4.7.2.12 pobierzNazwePliku()	19

Chapter 1

Class Index

1.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

BSTree	Klasa implementująca strukturę binarnego drzewa poszukiwań (BST)	5
FileManager	Klasa narzędziowa do obsługi operacji plikowych dla drzewa BST	8

Chapter 2

File Index

2.1 File List

Here is a list of all documented files with brief descriptions:

BSTree.cpp	Implementacja (definicje metod) klasy BSTree	11
BSTree.h	Deklaracja (interfejs) klasy binarnego drzewa poszukiwań (BSTree)	11
FileManager.cpp	Implementacja (definicje metod) klasy FileManager	13
FileManager.h	Deklaracja (interfejs) klasy FileManager	13
main.cpp	Główny plik programu, obsługa menu i integracja klas BSTree i FileManager	14

Chapter 3

Class Documentation

3.1 BSTree Class Reference

Klasa implementująca strukturę binarnego drzewa poszukiwań (BST).

```
#include <BSTree.h>
```

Public Member Functions

- [BSTree \(\)](#)
Konstruktor domyślny klasy `BSTree`. Tworzy puste drzewo (korzeń = nullptr).
- [~BSTree \(\)](#)
Destruktor klasy `BSTree`. Automatycznie wywołuje metodę `wyczysc()` w celu zwolnienia pamięci.
- [void dodaj \(int wartosc\)](#)
Dodaje nową wartość do drzewa BST.
- [void usun \(int wartosc\)](#)
Usuwa węzeł o podanej wartości z drzewa BST.
- [void wyczysc \(\)](#)
Usuwa wszystkie węzły z drzewa, zwalniając pamięć.
- [void znajdzSciezke \(int wartosc\)](#)
Wyszukuje ścieżkę od korzenia do węzła o podanej wartości. Ścieżka jest drukowana bezpośrednio do konsoli.
- [void zapiszDoTekstowego \(const std::string &nazwaPliku\)](#)
Zapisuje zawartość drzewa do pliku tekstowego. Wartości są zapisywane w kolejności Inorder (posortowane).
- [void wyswietl_preorder \(\)](#)
Wyświetla zawartość drzewa w konsoli w kolejności Preorder.
- [void wyswietl_inorder \(\)](#)
Wyświetla zawartość drzewa w konsoli w kolejności Inorder. (Wynik będzie posortowany rosnąco).
- [void wyswietl_postorder \(\)](#)
Wyświetla zawartość drzewa w konsoli w kolejności Postorder.
- [void wyswietlGraficznie \(\)](#)
Wyświetla strukturę drzewa w konsoli w formie "graficznej". (Drzewo jest obrócone o 90 stopni).

Friends

- class [FileManager](#)
Deklaracja przyjaźni z klasą `FileManager`. Umożliwia klasie `FileManager` dostęp do prywatnych pól (korzeń) i typów (Wezel) tej klasy w celu zapisu binarnego.

3.1.1 Detailed Description

Klasa implementująca strukturę binarnego drzewa poszukiwań (BST).

- Zarządza węzłami przechowującymi liczby całkowite, umożliwiając operacje takie jak dodawanie, usuwanie, przeszukiwanie i wyświetlanie.

3.1.2 Constructor & Destructor Documentation

3.1.2.1 BSTree()

```
BSTree::BSTree ()
```

Konstruktor domyślny klasy [BSTree](#). Tworzy puste drzewo (korzen = nullptr).

Konstruktor domyślny klasy [BSTree](#). Inicjalizuje drzewo jako puste (korzeń wskazuje na nullptr).

3.1.2.2 ~BSTree()

```
BSTree::~BSTree ()
```

Destruktor klasy [BSTree](#). Automatycznie wywołuje metodę [wyczysc\(\)](#) w celu zwolnienia pamięci.

Destruktor klasy [BSTree](#). Automatycznie zwalnia całą pamięć zajmowaną przez węzły drzewa, wywołując metodę [wyczysc\(\)](#).

3.1.3 Member Function Documentation

3.1.3.1 dodaj()

```
void BSTree::dodaj (
    int wartosc)
```

Dodaje nową wartość do drzewa BST.

Publiczna metoda dodająca nową wartość do drzewa. Jest "opakowaniem" dla rekurencyjnej funkcji [dodajHelper](#).

Parameters

wartosc	Liczba całkowita do wstawienia.
---------	---------------------------------

3.1.3.2 usun()

```
void BSTree::usun (
    int wartosc)
```

Usuwa węzeł o podanej wartości z drzewa BST.

Publiczna metoda usuwająca węzeł o podanej wartości z drzewa. Jest "opakowaniem" dla rekurencyjnej funkcji [usunHelper](#).

Parameters

wartosc	Liczba całkowita do usunięcia.
---------	--------------------------------

3.1.3.3 wyczysc()

```
void BSTree::wyczysc ()
```

Usuwa wszystkie węzły z drzewa, zwalniając pamięć.

Publiczna metoda usuwająca wszystkie węzły z drzewa. Drzewo staje się puste, a korzeń jest ustawiany na nullptr.

3.1.3.4 wyswietl_inorder()

```
void BSTree::wyswietl_inorder ()
```

Wyświetla zawartość drzewa w konsoli w kolejności Inorder. (Wynik będzie posortowany rosnąco).

Wyświetla drzewo w kolejności Inorder (posortowane).

3.1.3.5 wyswietl_postorder()

```
void BSTree::wyswietl_postorder ()
```

Wyświetla zawartość drzewa w konsoli w kolejności Postorder.

Wyświetla drzewo w kolejności Postorder.

3.1.3.6 wyswietl_preorder()

```
void BSTree::wyswietl_preorder ()
```

Wyświetla zawartość drzewa w konsoli w kolejności Preorder.

Wyświetla drzewo w kolejności Preorder.

3.1.3.7 wyswietlGraficznie()

```
void BSTree::wyswietlGraficznie ()
```

Wyświetla strukturę drzewa w konsoli w formie "graficznej". (Drzewo jest obrócone o 90 stopni).

Wyświetla strukturę drzewa "graficznie" w konsoli (obrócone).

3.1.3.8 zapiszDoTekstowego()

```
void BSTree::zapiszDoTekstowego (
    const std::string & nazwaPliku)
```

Zapisuje zawartość drzewa do pliku tekstowego. Wartości są zapisywane w kolejności Inorder (posortowane).

Publiczna metoda zapisująca drzewo do pliku tekstowego. Używa metody inorder (posortowanej) do zapisu.

Parameters

<i>nazwaPliku</i>	Nazwa pliku wyjściowego.
<i>nazwaPliku</i>	Nazwa pliku wyjściowego (np. "drzewo.txt").

3.1.3.9 `znajdzSciezke()`

```
void BSTree::znajdzSciezke (
    int wartosc)
```

Wyszukuje ścieżkę od korzenia do węzła o podanej wartości. Ścieżka jest drukowana bezpośrednio do konsoli.

Publiczna metoda szukająca ścieżki od korzenia do węzła. Wyświetla ścieżkę w konsoli.

Parameters

<i>wartosc</i>	Wartość, do której szukana jest ścieżka.
<i>wartosc</i>	Wartość węzła, do którego szukamy ścieżki.

The documentation for this class was generated from the following files:

- [BSTree.h](#)
- [BSTree.cpp](#)

3.2 FileManager Class Reference

Klasa narzędziowa do obsługi operacji plikowych dla drzewa BST.

```
#include <FileManager.h>
```

Public Member Functions

- void [wczytajTekstowyDoDrzewa](#) (`BSTree` &drzewo, const `std::string` &*nazwaPliku*)
Wczytuje liczby z pliku tekstowego i dodaje je do drzewa.
- void [zapiszBinarnie](#) (`BSTree` &drzewo, const `std::string` &*nazwaPliku*)
Zapisuje całą strukturę drzewa do pliku binarnego.
- `BSTree` [wczytajBinarnie](#) (const `std::string` &*nazwaPliku*)
Wczytuje strukturę drzewa z pliku binarnego.

3.2.1 Detailed Description

Klasa narzędziowa do obsługi operacji plikowych dla drzewa BST.

- Zapewnia metody do serializacji (zapisu) i deserializacji (odczytu) drzewa do/z formatu tekstowego i binarnego.

3.2.2 Member Function Documentation

3.2.2.1 wczytajBinarnie()

```
BSTree FileManager::wczytajBinarnie (
    const std::string & nazwaPliku)
```

Wczytuje strukturę drzewa z pliku binarnego.

- Tworzy i zwraca *nowy* obiekt **BSTree**, odtworzony z pliku binarnego.
- Zakłada, że plik został zapisany w kolejności Preorder.

Parameters

<i>nazwaPliku</i>	Nazwa pliku binarnego do wczytania.
-------------------	-------------------------------------

Returns

Nowy obiekt **BSTree** odtworzony z pliku.

Otwiera plik binarny, czyta liczby całkowite jedna po drugiej i buduje *nowe* drzewo, dodając je w kolejności odczytu. Działa poprawnie tylko wtedy, gdy plik był zapisany w kolejności Preorder.

Parameters

<i>nazwaPliku</i>	Nazwa pliku binarnego do wczytania.
-------------------	-------------------------------------

Returns

Nowy obiekt **BSTree** odtworzony z pliku. Zwraca puste drzewo, jeśli otwarcie pliku się nie powiodło.

3.2.2.2 wczytajTekstowyDoDrzewa()

```
void FileManager::wczytajTekstowyDoDrzewa (
    BSTree & drzewo,
    const std::string & nazwaPliku)
```

Wczytuje liczby z pliku tekstowego i dodaje je do drzewa.

- Otwiera plik tekstowy i dodaje każdą znalezioną liczbę do istniejącego obiektu drzewa.

Parameters

<i>drzewo</i>	Referencja do obiektu BSTree , który ma być zmodyfikowany.
<i>nazwaPliku</i>	Nazwa pliku .txt do wczytania.

Otwiera plik o podanej nazwie, czyta liczby całkowite (oddzielone białymi znakami) i dodaje je do istniejącego obiektu drzewa za pomocą metody `drzewo.doda j ()`.

Parameters

<i>drzewo</i>	Referencja do obiektu BSTree , który ma być zmodyfikowany.
<i>nazwaPliku</i>	Nazwa pliku .txt do wczytania.

3.2.2.3 [zapiszBinarnie\(\)](#)

```
void FileManager::zapiszBinarnie (
    BSTree & drzewo,
    const std::string & nazwaPliku)
```

Zapisuje całą strukturę drzewa do pliku binarnego.

Publiczna metoda zapisująca drzewo do pliku binarnego.

- Wykorzystuje prywatną funkcję pomocniczą do rekurencyjnego zapisu w kolejności Preorder.

Parameters

<i>drzewo</i>	Referencja do drzewa, które ma być zapisane.
<i>nazwaPliku</i>	Nazwa pliku wyjściowego (np. "drzewo.bin").

Otwiera plik binarny do zapisu i rozpoczyna rekurencyjny proces zapisu, wywołując [zapiszBinarnie<->Helper](#). Wymaga przyjaźni z klasą [BSTree](#), aby uzyskać dostęp do korzenia.

Parameters

<i>drzewo</i>	Referencja do drzewa, które ma być zapisane.
<i>nazwaPliku</i>	Nazwa pliku wyjściowego (np. "drzewo.bin").

The documentation for this class was generated from the following files:

- [FileManager.h](#)
- [FileManager.cpp](#)

Chapter 4

File Documentation

4.1 BSTree.cpp File Reference

Implementacja (definicje metod) klasy [BSTree](#).

```
#include "BSTree.h"
#include <iostream>
#include <fstream>
#include <string>
```

4.1.1 Detailed Description

Implementacja (definicje metod) klasy [BSTree](#).

4.2 BSTree.h File Reference

Deklaracja (interfejs) klasy binarnego drzewa poszukiwań ([BSTree](#)).

```
#include <string>
#include <iostream>
```

Classes

- class [BSTree](#)

Klasa implementująca strukturę binarnego drzewa poszukiwań (BST).

4.2.1 Detailed Description

Deklaracja (interfejs) klasy binarnego drzewa poszukiwań ([BSTree](#)).

4.3 BSTree.h

[Go to the documentation of this file.](#)

```
00001
00005
00006 #ifndef BSTREE_H
00007 #define BSTREE_H
00008
00009 #include <string>
00010 #include <iostream>
00011
00012 class FileManager;
00013
00020 class BSTree {
00021
00027 friend class FileManager;
00028
00029 private:
00030
00036     struct Wezel {
00038         int dane;
00040         Wezel* lewy;
00042         Wezel* prawy;
00043
00044         Wezel(int wartosc) {
00045             dane = wartosc;
00046             lewy = nullptr;
00047             prawy = nullptr;
00048         }
00049     };
00050
00051     Wezel* korzen;
00052
00053
00054     Wezel* dodajHelper(Wezel* wezel, int wartosc);
00055
00056     Wezel* usunHelper(Wezel* wezel, int wartosc);
00057
00058     Wezel* znajdzMin(Wezel* wezel);
00059
00060     void wyczyscHelper(Wezel* wezel);
00061
00062     void preorderHelper(Wezel* wezel);
00063
00064     void inorderHelper(Wezel* wezel);
00065
00066     void postorderHelper(Wezel* wezel);
00067
00068     bool znajdzSciezkeHelper(Wezel* wezel, int wartosc);
00069
00070     void zapiszHelper(Wezel* wezel, std::ostream& plik);
00071
00072     void wyswietlGraficznieHelper(Wezel* wezel, std::string wciecie, bool czyPrawy);
00073
00074
00075 public:
00076     BSTree();
00077
00078     ~BSTree();
00079
00080     void dodaj(int wartosc);
00081
00082     void usun(int wartosc);
00083
00084     void wyczysc();
00085
00086     void znajdzSciezke(int wartosc);
00087
00088     void zapiszDoTekstowego(const std::string& nazwaPliku);
00089
00090
00091     void wyswietl_preorder();
00092
00093     void wyswietl_inorder();
00094
00095     void wyswietl_postorder();
00096
00097     void wyswietl_Graficznie();
00098
00099 };
00100
00101 #endif // BSTREE_H
```

4.4 FileManager.cpp File Reference

Implementacja (definicje metod) klasy [FileManager](#).

```
#include "FileManager.h"
#include <fstream>
#include <iostream>
```

4.4.1 Detailed Description

Implementacja (definicje metod) klasy [FileManager](#).

Zawiera logikę wczytywania i zapisywania drzewa BST do plików tekstowych oraz binarnych.

4.5 FileManager.h File Reference

Deklaracja (interfejs) klasy [FileManager](#).

```
#include <string>
#include "BSTree.h"
#include <iostream>
```

Classes

- class [FileManager](#)

Klasa narzędziowa do obsługi operacji plikowych dla drzewa BST.

4.5.1 Detailed Description

Deklaracja (interfejs) klasy [FileManager](#).

Klasa ta odpowiada za operacje wejścia/wyjścia, w tym wczytywanie i zapisywanie drzewa BST do plików.

4.6 FileManager.h

[Go to the documentation of this file.](#)

```
00001
00007
00008 #ifndef FILEMANAGER_H
00009 #define FILEMANAGER_H
00010
00011 #include <string>
00012 #include "BSTree.h"
00013 #include <iostream>
00014
00021 class FileManager {
00022
00023 private:
00031     void zapiszBinarnieHelper(BSTree::Wezel* wezel, std::ostream& plik);
00032
00033 public:
00041     void wczytajTekstowyDoDrzewa(BSTree& drzewo, const std::string& nazwaPliku);
00042
00050     void zapiszBinarnie(BSTree& drzewo, const std::string& nazwaPliku);
00051
00059     BSTree wczytajBinarnie(const std::string& nazwaPliku);
00060
00061 };
00062
00063 #endif // FILEMANAGER_H
```

4.7 main.cpp File Reference

Główny plik programu, obsługa menu i integracja klas [BSTree](#) i [FileManager](#).

```
#include <iostream>
#include <string>
#include <limits>
#include "BSTree.h"
#include "FileManager.h"
```

Functions

- void [wyswietlMenu](#) ()
Wyświetla w konsoli wszystkie dostępne opcje menu.
- int [pobierzLiczbe](#) (const std::string &prompt)
Bezpiecznie pobiera od użytkownika liczbę całkowitą.
- std::string [pobierzNazwePliku](#) (const std::string &prompt)
Pobiera od użytkownika nazwę pliku (jako pojedyncze słowo).
- void [op_dodaj](#) (BSTree &drzewo)
Obsługuje logikę opcji menu '1. Dodaj element'.
- void [op_usun](#) (BSTree &drzewo)
Obsługuje logikę opcji menu '2. Usuń element'.
- void [op_wyswietl](#) (BSTree &drzewo)
Obsługuje logikę opcji menu '3. Wyświetl drzewo'.
- void [op_znajdzSciezke](#) (BSTree &drzewo)
Obsługuje logikę opcji menu '4. Znajdź ścieżkę do elementu'.
- void [op_wyczysc](#) (BSTree &drzewo)
Obsługuje logikę opcji menu '5. Usuń całe drzewo'.
- void [op_zapiszTekst](#) (BSTree &drzewo)
Obsługuje logikę opcji menu '7. Zapisz drzewo do pliku tekstowego'.
- void [op_wczytajTekst](#) (BSTree &drzewo, FileManager &fm)
Obsługuje logikę opcji menu '6. Wczytaj drzewo z pliku tekstowego'.
- void [op_zapiszBin](#) (BSTree &drzewo, FileManager &fm)
Obsługuje logikę opcji menu '8. Zapisz drzewo do pliku binarnego'.
- void [op_wczytajBin](#) (BSTree &drzewo, FileManager &fm)
Obsługuje logikę opcji menu '9. Wczytaj drzewo z pliku binarnego'.
- int [main](#) ()
Główna funkcja programu i punkt startowy.

4.7.1 Detailed Description

Główny plik programu, obsługa menu i integracja klas [BSTree](#) i [FileManager](#).

4.7.2 Function Documentation

4.7.2.1 main()

```
int main ()
```

Główna funkcja programu i punkt startowy.

- Inicjalizuje obiekty `BSTree` i `FileManager`, a następnie uruchamia główną pętlę menu, która obsługuje interakcje z użytkownikiem.

Returns

- 0 po poprawnym zakończeniu programu (wybór opcji '0').

4.7.2.2 op_dodaj()

```
void op_dodaj (
    BSTree & drzewo)
```

Obsługuje logikę opcji menu '1. Dodaj element'.

- Prosi użytkownika o wartość i wywołuje metodę `doda j` na drzewie.

-

Parameters

<code>drzewo</code>	Referencja do obiektu drzewa, do którego dodajemy.
---------------------	--

4.7.2.3 op_usun()

```
void op_usun (
    BSTree & drzewo)
```

Obsługuje logikę opcji menu '2. Usuń element'.

- Prosi użytkownika o wartość i wywołuje metodę `usun` na drzewie.

-

Parameters

<code>drzewo</code>	Referencja do obiektu drzewa, z którego usuwamy.
---------------------	--

4.7.2.4 op_wczytajBin()

```
void op_wczytajBin (
    BSTree & drzewo,
    FileManager & fm)
```

Obsługuje logikę opcji menu '9. Wczytaj drzewo z pliku binarnego'.

- Prosi o nazwę pliku i wywołuje metodę `wczyta jBinarnie`. Następnie nadpisuje istniejący obiekt drzewa nowym, wczytanym drzewem.

-

Parameters

<i>drzewo</i>	Referencja do drzewa, które zostanie nadpisane.
<i>fm</i>	Referencja do obiektu FileManager .

4.7.2.5 op_wczytajTekst()

```
void op_wczytajTekst (
    BSTree & drzewo,
    FileManager & fm)
```

Obsługuje logikę opcji menu '6. Wczytaj drzewo z pliku tekstowego'.

- Prosi o nazwę pliku i wywołuje metodę `wczytajTekstowyDoDrzewa` na managerze plików.

-

Parameters

<i>drzewo</i>	Referencja do drzewa, do którego będą dodane elementy.
<i>fm</i>	Referencja do obiektu FileManager .

4.7.2.6 op_wyczysc()

```
void op_wyczysc (
    BSTree & drzewo)
```

Obsługuje logikę opcji menu '5. Usuń całe drzewo'.

- Wywołuje metodę `wyczysc` na drzewie.

-

Parameters

<i>drzewo</i>	Referencja do drzewa, które ma być wyczyszczone.
---------------	--

4.7.2.7 op_wyswietl()

```
void op_wyswietl (
    BSTree & drzewo)
```

Obsługuje logikę opcji menu '3. Wyświetl drzewo'.

- Wyświetla pod-menu wyboru metody wyświetlania (Preorder, Inorder, itd.) i wywołuje odpowiednią metodę na obiekcie drzewa.

-

Parameters

<code>drzewo</code>	Referencja do obiektu drzewa do wyświetlenia.
---------------------	---

4.7.2.8 `op_zapiszBin()`

```
void op_zapiszBin (
    BSTree & drzewo,
    FileManager & fm)
```

Obsługuje logikę opcji menu '8. Zapisz drzewo do pliku binarnego'.

- Prosi o nazwę pliku i wywołuje metodę `zapiszBinarnie` na managerze plików.
- **Parameters**

<code>drzewo</code>	Referencja do drzewa, które ma być zapisane.
<code>fm</code>	Referencja do obiektu <code>FileManager</code> .

4.7.2.9 `op_zapiszTekst()`

```
void op_zapiszTekst (
    BSTree & drzewo)
```

Obsługuje logikę opcji menu '7. Zapisz drzewo do pliku tekstowego'.

- Prosi o nazwę pliku i wywołuje metodę `zapiszDoTekstowego` na drzewie.
- **Parameters**

<code>drzewo</code>	Referencja do drzewa, które ma być zapisane.
---------------------	--

4.7.2.10 `op_znajdzSciezke()`

```
void op_znajdzSciezke (
    BSTree & drzewo)
```

Obsługuje logikę opcji menu '4. Znajdź ścieżkę do elementu'.

- Prosi o wartość i wywołuje metodę `znajdzSciezke` na drzewie.
- **Parameters**

<code>drzewo</code>	Referencja do obiektu drzewa, w którym szukamy.
---------------------	---

4.7.2.11 pobierzLiczbe()

```
int pobierzLiczbe (
    const std::string & prompt)
```

Bezpieczne pobiera od użytkownika liczbę całkowitą.

- Posiada walidację (pętlę), która wymusza podanie poprawnej liczby i ignoruje błędne dane wejściowe (np. tekst).

- **Parameters**

<i>prompt</i>	Tekst zachęty wyświetlany użytkownikowi (np. "Podaj liczbę: ").
---------------	---

Returns

Poprawnie wczytana liczba całkowita.

4.7.2.12 pobierzNazwePliku()

```
std::string pobierzNazwePliku (
    const std::string & prompt)
```

Pobiera od użytkownika nazwę pliku (jako pojedyncze słowo).

- **Parameters**

<i>prompt</i>	Tekst zachęty (np. "Podaj nazwę pliku: ").
---------------	--

Returns

Nazwa pliku jako std::string.

Index

~BSTree
 BSTree, 6

BSTree, 5
 ~BSTree, 6
 BSTree, 6
 dodaj, 6
 usun, 6
 wyczysc, 7
 wyswietl_inorder, 7
 wyswietl_postorder, 7
 wyswietl_preorder, 7
 wyswietlGraficznie, 7
 zapiszDoTekstowego, 7
 znajdzSciezke, 8

BSTree.cpp, 11

BSTree.h, 11

dodaj
 BSTree, 6

FileManager, 8
 wczytajBinarnie, 9
 wczytajTekstowyDoDrzewa, 9
 zapiszBinarnie, 10

FileManager.cpp, 13

FileManager.h, 13

main
 main.cpp, 15

main.cpp, 14
 main, 15
 op_dodaj, 15
 op_usun, 15
 op_wczytajBin, 15
 op_wczytajTekst, 16
 op_wyczysc, 16
 op_wyswietl, 16
 op_zapiszBin, 17
 op_zapiszTekst, 17
 op_znajdzSciezke, 17
 pobierzLiczbe, 17
 pobierzNazwePliku, 19

op_dodaj
 main.cpp, 15

op_usun
 main.cpp, 15

op_wczytajBin
 main.cpp, 15

op_wczytajTekst

main.cpp, 16
op_wyczysc
 main.cpp, 16

op_wyswietl
 main.cpp, 16

op_zapiszBin
 main.cpp, 17

op_zapiszTekst
 main.cpp, 17

op_znajdzSciezke
 main.cpp, 17

pobierzLiczbe
 main.cpp, 17

pobierzNazwePliku
 main.cpp, 19

usun
 BSTree, 6

wczytajBinarnie
 FileManager, 9

wczytajTekstowyDoDrzewa
 FileManager, 9

wyczysc
 BSTree, 7

wyswietl_inorder
 BSTree, 7

wyswietl_postorder
 BSTree, 7

wyswietl_preorder
 BSTree, 7

wyswietlGraficznie
 BSTree, 7

zapiszBinarnie
 FileManager, 10

zapiszDoTekstowego
 BSTree, 7

znajdzSciezke
 BSTree, 8